

**SYSTEM, METHOD AND COMPUTER SOFTWARE
PRODUCTS FOR NETWORK FIREWALL FAST POLICY LOOK-UP**

5 This Application claims a Priority Date of July 31, 2000, benefited from a previously filed Provisional Application 60/221,823 filed on July 31, 2000 by the same Applicant of this Patent Application.

BACKGROUND OF THE INVENTION

10 **1. Field of the Invention**

The present invention relates to computer network security. More particularly, this invention is related to firewall, i.e., a combination of computer hardware and software for selectively accepting network data communications and
15 rejecting unacceptable data transmissions to safeguard a computer network based on a predefined policy table.

2. Descriptions of the Reference Art

20 As network communications become more wide spread through the use of the Internet systems, many technical challenges are encountered by those of ordinary skill in the art to deal with the issues of network security. One specific challenge is to carry out the tasks of differentiating legitimate and illegitimate accesses to a protected network system effectively and expeditiously. As the
25 amount of data transmitted over the Internet and the sources and destinations of the data transmissions are increased exponentially, the speed and accuracy in carrying out the tasks of legitimacy differentiation becomes critically important. On the one-hand higher speed is required in order to process large of data transmissions. On the other hand, due to the open and unrestricted nature of transmitting data to
30 any and all designated destinations over the Internet, all network systems now become more vulnerable and exposed to illegitimate accesses and attacks.

In a general term, an Internet is a network of networks with a global collection of interconnected local, mid-level, and wide-area networks that use the
35 Internet Protocol (IP) as the network layer protocol. Whereas the Internet embraces

many local- and wide-area networks, a given local- or wide-area network may or may not form part of the Internet. For purposes of the present specification, a "wide-area network" (WAN) is a network that links at least two LANs over a wide geographical area via one or more dedicated connections. The public switched telephone network is an example of a wide- area network. A local-area network (LAN) is a network that takes advantage of the proximity of computers to typically offer relatively efficient, higher speed communications than wide-area networks. In addition, a network may use the same underlying technologies as the Internet. Such a network is referred to herein as an "Intranet," an internal network based on Internet standards. Because the Internet has become the most pervasive and popularly employed open networking standard, significant economic benefits are achieved by applying a same Internet standard in the internal networks. For these reasons, corporate Intranets have become a strong driving force in the marketplace of network products and services.

As the Internet and its underlying technologies have become increasingly familiar, attention has become focused on Internet security and computer network security in general. With unprecedented access to information, it has also come unprecedented opportunities to gain unauthorized access to data, change data, destroy data, make unauthorized use of computer resources, interfere with the intended use of computer resources, etc. As experience has shown, the frontier of cyber-space has its share of scofflaws, resulting in increased efforts to protect the data, resources, and reputations of those embracing Intranets and the Internet. Firewalls are intended to shield data and resources from the potential ravages of computer network intruders. In essence, a firewall functions as a mechanism, which monitors and controls the flow of data between two networks. All communications, e.g., data packets, which flow between the networks in either direction, must pass through the firewall; otherwise, security is circumvented. The firewall selectively permits the communications to pass from one network to the other, to provide bi-directional security.

Ideally, a firewall would be able to prevent any and all security breaches and attacks. Although absolute security is indeed a goal to be sought after, due to many variables (e.g., physical intrusion into the physical plant) it may be difficult to achieve. However, in many instances, it is of equal if not greater importance to be

alerted to an attack so that measures may be taken to thwart the attack or render it harmless, and to avoid future attacks of the same kind. Hence a firewall, in addition to security, should provide timely information that enables attacks to be detected. Firewalls have typically relied on some combination of two techniques affording network protection: packet filtering and proxy services.

Packet filtering is the action a firewall takes to selectively control the flow of data to and from a network. Packet filters allow or block packets, usually while routing them from one network to another (often from the Internet to an internal network and vice versa). To accomplish packet filtering, a network administrator establishes a set of rules that specify what types of packets (e.g., those to or from a particular IP address or port) are to be allowed to pass and what types are to be blocked. Packet filtering may occur in a router, in a bridge, or on an individual host computer.

Packet filters are typically configured in a "default permit stance", i.e., that which is not expressly prohibited is permitted. In order for a packet filter to prohibit potentially harmful traffic, it must know what the constituent packets of that traffic look like. However, it is virtually impossible to catalogue all the various types of potentially harmful packets and to distinguish them from benign packet traffic. The filtering function required to do so is too complex. Hence, while most packet filters may be effective in dealing with the most common types of network security threats, this methodology presents many chinks that an experienced hacker may exploit. The level of security afforded by packet filtering, therefore, leaves much to be desired.

Recently, a further network security technique termed "stateful inspection" has emerged. Stateful inspection performs packet filtering not on the basis of a single packet, but on the basis of some historical window of packets on the same port. Although stateful inspection may enhance the level of security achievable using packet filtering, it is as yet relatively unproven. Furthermore, although an historical window of packets may enable the filter to more accurately identify harmful packets, the filter must still know what it is looking for. Building a filter with sufficient intelligence to deal with the almost infinite variety of possible packets and packet sequences is liable to prove an exceedingly difficult task.

The other principal methodology used in present-day firewalls is proxies. In order to describe prior-art proxy-based firewalls, some further definitions are required. A "node" is an entity that participates in network communications. A sub-network is a portion of a network or a physically independent network that may share network addresses with other portions of the network. An intermediate system is a node that is connected to more than one subnetwork and that has the role of a router for forwarding data from one subnetwork to the other.

A proxy is a program, running on an intermediate system, that deals with servers (e.g., Web servers, FTP servers, etc.) on behalf of clients. Clients, e.g. computer applications that are attempting to communicate with a network that is protected by a firewall, send requests for connections to proxy-based intermediate systems. Proxy-based intermediate Systems relay approved client requests to target servers and relay answers back to clients.

Proxies require either custom software (i.e., proxy-aware applications) or custom user procedures in order to establish a connection. Using custom software for proxying presents several problems. Appropriate custom client software is often available only for certain platforms and the software available for a particular platform may not be the software that users prefer. Furthermore, using custom client software, users must perform extra manual configuration to direct the software to contact the proxy on the intermediate system. With the custom procedure approach, the user tells the client to connect to the proxy and then tells the proxy which host to connect to. Typically, the user will first enter the name of a firewall that the user wishes to connect through. The firewall will then prompt the user for the name of the remote host the user wishes to connect to. Although this procedure is relatively simple in the case of a connection that traverses only a single firewall, as network systems grow in complexity, a connection may traverse several firewalls. Establishing a proxied connection in such a situation starts to become a confusing maze, and a significant burden to the user, since the user must know the route the connection is to take. Furthermore, since proxies must typically prompt the user or the client software for a destination using a specific protocol, they are protocol-specific. Separate proxies are therefore required for each protocol that is to be used.

In general, network firewalls employ filter rules or policies to police network communication. In such implementation, a data packet is examined and checked with fire filter policy rules. In essence, the policy lookup in the network firewall is to find an efficient way to map a four-dimensional space DA, SA, DP, SP, to one dimension policy space. Historically, most firewalls use linear search algorithms. These algorithms are very time consuming and without upper bound of searching time the searching time increase linearly as the Policy List growing.

Therefore, a need still exists in the art to provide effective method to enable a person of ordinary skill in the art to effectively differentiate allowable/disallowable network accesses with high speed and accuracy to resolve these difficulties. Specifically, the method must be conveniently adaptable to computer implementation. It is further desirable that the efficiency and accuracy can be indexed as ordered lists for conveniently sorted, updated, and reorganized when there are configuration changes of a network systems.

SUMMARY OF THE PRESENT INVENTION

It is the object of the present invention to provide a new and improved method to effectively identify a policy-table allowable data communication received from a network by employing a multiple-dimensional spatial indexing and mapping methods for speed and accuracy improvements. By systematically converting address and port numbers of a policy table into sequential numbers and by mapping the sequential number to policy entry-counters, lookup efficiency is greatly improved through traveling down binary tree of port and address sequential numbers. Additionally, performance of actual policy-number identification is made through mapping via consolidated and indexed multiple dimensional spaces. Therefore, the difficulties and limitations as discussed above commonly encountered in the conventional techniques are resolved.

In one aspect of the invention, a fast policy lookup (FPL) process is implemented. The use of the FPL in computer systems and firewall software products improves the speed of policy (rule) look-up because the table lookup is now carried out in a systematic way according to an ordered sequence. In a

5 preferred embodiment, the FPL divides the two IP addresses (DA, SA) and the two Port Numbers (DP, SP) spaces into non-overlapped segments according to the address book. More precisely, according the addresses used in the Policy List and the Service Type List. The original four-dimensional space is now reduced to a two-dimensional space wherein the two-dimensional space is also indexed according to a policy table entry number and then combined into a two-dimensional policy table. Consolidations through index mapping of lists defined in multi-dimensional spaces are employed to simplify the table lookup processes.

10 A preferred embodiment of this invention discloses a method for processing a policy table comprising a plurality of policy-table entries. Each entry comprises data for defining a plurality of destination address ranges, a source address ranges, a destination port group and a source port group. The method includes steps of A) assigning an ordered sequence number as a policy-table entry counter ip to each of
15 the policy table entries. B) Fragmenting the destination address ranges and the source address ranges listed in the policy table entries into a plurality of a sequentially-ordered destination address segments and source address segments respectively and each segment is assigned with a sequential segment number thus generating a set of source address sequence numbers (SASN) and a set of
20 destination address sequence numbers (DASN). C) forming a source-destination address mapping table (SDAMT) comprising a plurality of SDAMT table entries for each pair of SASN and DASN wherein each of the SDAMT table entries is provided with a policy-table entry counter ip corresponding to a first policy table entry wherein the SASN and DASN being listed. D) fragmenting the destination
25 port groups and the source port groups listed in the policy table entries into a plurality of a sequentially-ordered destination port segments and source port segments respectively and each segment is assigned with a sequential segment number thus generating a set of source port sequence numbers (SPSN) and a set of destination port sequence numbers (DPSN). And E) forming a source-destination
30 port mapping table (SDPMT) comprising a plurality of SDPMT table entries for each pair of SPSN and DPSN wherein each of the SDPMT table entries is provided with a policy-table entry counter ip corresponding to a first policy table entry wherein the SPSN and DPSN being listed.

The invention also discloses a method for processing a table comprising a plurality of table entries with each entry providing data for defining a plurality of multiple-dimensional spaces. The method includes steps of A) assigning an ordered sequence number as a table entry counter ip to each of the table entries. B) 5 Fragmenting the multiple-dimensional spaces into order spatial ranges and assigned each of the spatial ranges with a sequential spatial range-numbers. C) Forming multiple-dimensional range-spaces by employing the sequential spatial range-numbers as coordinates and assigning an associated table entry counter ip to each block defined by the spatial range-number coordinates for providing an index for correlating each of the sequential spatial range-numbers to the each of the table 10 entry.

These and other objects and advantages of the present invention will no doubt become obvious to those of ordinary skill in the art after having read the 15 following detailed descriptions of the preferred embodiment that is illustrated in the various drawing figures.

BRIEF DESCRIPTION OF THE DRAWINGS

20 Figure 1 is a flow chart showing the processes of a fast policy lookup method disclosed by this invention;

Figure 2 shows Internet Protocol address and port number segmentation;

25 Figure 3 shows SDAMT-source and destination address mapping table;

Figure 4 shows SDPMT-source destination port mapping table.

30 Figures 5A to 5C illustrate process of employing the table entries of the SDAMT, and SDPMT to form a policy mapping table of Fig. 5C.

DETAILED DESCRIPTION OF THE METHOD

35 Reference will now be made in detail to the preferred embodiments of the invention while the invention will be described in conjunction with the preferred

embodiments, however, it is not the intent of the Applicant to limit the scope of the invention to these embodiments. On the contrary, the scope of the invention is intended to cover alternatives, modifications and equivalents, which may be included within the spirit and scope of the invention. As will be appreciated by one of skill in the art, the present invention may be embodied as methods, systems or computer software program products. Software written within the scope of the present invention may be stored in some form of computer readable medium, such as memory, or hard-drive, CD-ROM. Furthermore, the software of the invention may be transmitted over a network and executed by a processor in a remote location. The software may also be embedded in the computer readable medium of hardware, such as a network gateway device or a network card.

Referring to Fig. 1 for carrying out a policy-table lookup process according to the method of this invention. A policy-table includes a plurality of policy entries defining the acceptable incoming packets allowable for the firewall-protected network to receiving into the system as input packets. Each of these policy entries includes three sets of information: 1) a source subnet defined by a range of source IP addresses (SA) and a destination subnet defined by a range of destination IP address (DA). 2) A source port group defined by a range of port numbers (SP) and a destination port group defined by a range of port numbers (DP). And, 3) a protocol type. The protocol type has several choices, e.g., TCP/IP or UDP/IP. For the purpose of this invention, the protocol types provided in the entries of the policy-table are irrelevant when carrying out the table-lookup process for differentiating the policy-table allowable packets.

Referring to Fig. 1 again, the policy-table lookup process begins (step 100) with a process to first organize the policy table into multiple dimensional spaces for the purpose of establishing an indexing system related to each entry of the policy table. The process begins by sequentially examining every entry of the policy table. A policy table generally includes a list of policy entries and each entry is typically represented by:

{<Destination Subnet, Source Subnet>, <Destination Port, Source Port>, Protocol type, → Actions}

A range of IP addresses defines the subnet and a range of port numbers defines the port group as that provided in each policy entry. In common network configuration, the source and destination addresses are defined by a 32-bit word and the source and port numbers are defined by a 16-bit word. Therefore, there can be 2^{32} source and destination addresses and 2^{16} source and destination ports.

The policy table is organized into indexed tables by a step of fragmentation of the Internet Protocol (IP) source address (SA) into non-overlapping segments (step 105) and fragmentation of the destination addresses (DA) into non-overlapping segments (step 110). Referring to Fig. 2 for the fragmentation process of the one-dimensional array of ranges of source or destination addresses or port numbers. Each entry of the policy table is examined by looking at the range of source IP addresses defined by a minimum and maximum source IP addresses. These maximum and minimum IP addresses of the source IP address-range are selected as segment separation points as that shown in Fig. 2. All the minimum and maximum addresses for all the ranges provided in the policy table are marked as separation points over the one-dimensional axis thus forming a plurality of non-overlapping segments over a one-dimension space. This same process is performed for the destination IP addresses. As that shown in Fig. 2, each segment is assigned a segment number according to an ascending sequential order start from segment number 0. A two dimensional space, represented by a two-dimensional source-destination address mapping table (SDAMT) is formed using the SA segment sequential number (SASN) as the X-axis and the DA segment sequential number (DASN) as the Y-axis (step 115). As that shown in Fig. 3, each entry of this two-dimensional SDAMPT table that represents an index value for a {SASN, DASN} pair. Identical steps are carried out by examining the source port group and the destination port group as provided in each entry of the policy table to first fragmentize and define the source port sequential number (SPSN) and destination port sequential number (DPSN) (steps 120 and 125). Then a two-dimensional source-destination port mapping table (SDPMT) is formed corresponding to a two-dimensional space with source-port sequential number (SPSN) and destination-port sequential number (DPSN) representing the X-axis and Y-axis respectively. As that shown in Fig. 4, each entry of this two-dimensional SDPMT table represents an index value for a corresponding {SPSN, DPSN} pair.

Referring to Fig. 1 again, each entry of the policy table is assigned a policy entry counter $ip = 0, 1, 2, 3, \dots, N$, according to an ascending sequential order starting from zero (step 135) where N is the total number of policy entries in the policy table. The process continues by assign an policy entry counter ip to each table entry corresponding to every $\{SASN, DASN\}$ pair in the source-destination address mapping table and each table entry corresponding every $\{SPSN, DPSN\}$ pair in the source-destination port mapping table (SDPMT) (step 140). All the table entries are initially registered as "unused" before the policy entry counter ip is entered in either the SDAMT or the SDPMT tables, and each table entry in either of these two tables is entered only with the first ip counter. Once a policy entry counter ip is entered for a table entry, that table entry in either the SDAMT or SDPMT tables is assigned with one unique ip counter and will not be changed unless overwritten by other procedure when there are changes made to the policy table. A mapping process is then carried out to transform from the four dimensional space defined by four entries of ip in four tables, i.e., SDAMT and SDPMT, to another two dimensional space represented by a policy mapping table (PMT) (step 145).

Referring to Figs. 5A to 5C for an example for illustrating the mapping process to construct the policy-mapping table. Figs. 5A and 5B shows the SDAMT and SDPMT entries at the time when the processes for constructing these two tables are completed for the policy entry counter $ip=4$. For policy-entry counter $ip=1$, examining Figs. 5A and 5B, there is only one combination, i.e., $\{1, 1\}$. An ip counter number, i.e., $ip = 1$, is entered into the slot $\{1, 1\}$ of the policy mapping table (PMT). For $ip = 2$, there are possible combinations of $\{1, 2\}$ and $\{2, 2\}$. An ip counter number, i.e., $ip = 2$, is entered into the slot $\{3, 1\}$, and $\{2, 2\}$ of the policy mapping table (PMT). For $ip = 3$ there are possible combinations of $\{3, 1\}$ and $\{3, 3\}$. An ip counter number, i.e., $ip = 3$, is entered into the slot $\{3, 1\}$, and $\{3, 3\}$ of the policy mapping table (PMT). For $ip = 4$, the possible combinations are $\{4, 2\}$ and $\{4, 4\}$. An ip counter number, i.e., $ip = 4$, is entered into the slot $\{4, 2\}$, and $\{4, 4\}$ of the policy mapping table (PMT). The X-Y coordinates on the PMT table are therefore generated by combining the policy entry counters from the source-destination address mapping table (SDAMT) as the X-coordinate, and the policy entry counters from the source-destination port mapping table (SDPMT) as the Y-coordinate for all policy entry counter $ip = 1, 2, 3, \dots, N$, a policy mapping table is formed. A two two-dimensional tables are mapped into a two dimensional policy

mapping table as that illustrated in Fig. 5C.

Referring back to Fig. 1 again, for the purpose of effectively conducting a “fast policy lookup” process, four “binary trees” are structured (step 150). These four binary trees are a source address tree, a destination address tree, a source-port tree and destination-port tree. Suppose that there are N source and destination addresses and M source and destination port, the process generally start from a root of represented by a source/destination address sequence number of N/2 and source/destination port number of M/2. Each binary tree starts with a root N/2 or M/2, each having two branches having the source-destination address and port sequence numbers of $[(N/2-1), (N/2+1)]$ and $[(M/2-1), (M/2+1)]$. In receiving an incoming packet, the header of the packet is parsed to get the source/destination addresses and source/destination port number (step 155). These address and port number are then applied to travel down the four binary trees to find the source/destination address sequence numbers, i.e., SASN and DASN, and the source-destination port sequence number, i.e., SPSN and DPSN (step 160). Using the SASN and DASN as X-Y coordinates, a policy entry counter $ip(A)$ is determined from the SDAMT as that shown in Fig. 5A. Using the SPSN and DPSN as X-Y coordinates, a policy entry counter $ip(P)$ is determined from the SDPMT as that shown in Fig. 5B (step 165). These two policy entry counter numbers $ip(A)$ and $ip(P)$ are then used as X-Y coordinates to lookup the final policy entry counter number from the policy mapping table as that shown in Fig. 5C (step 170).

To further summarize the processing steps of this invention, the following descriptions present a framework to outline a processing flow of the invention.

First, two tables are generated:

SDAMT-Source and Destination Address Mapping Table

SDPMT-Source and Destination Port Mapping Table

Second, the 2-dimension space resulted from the previous step is transformed to the final policy space by looking up the third table:

PMT-Policy Mapping Table

There are many ways to map a given IP address to a segment. In one embodiment, this is achieved by maintaining a balanced binary tree. For the port number mapping, a direct table (65536 in size) lookup may be more efficient and feasible in some

embodiments.

IP Address Fragmentation

5 IP address fragmentation should be done for both source IP address space and destination IP address space respectively. The methods for carrying out IP address space fragmentation are exactly the same. A fragmentation of the source IP address space is described below as an example.

10 For each source sub-net appeared in the policy list, we use its two boundary IP addresses as the separating point in the IP space, keep doing this for every entries in the Policy List. When this is finished, we assign each segment a sequence number in the ascend order starting from 0. (See figure 1)

15 Port Number Fragmentation

The principle of Port Number Fragmentation is quite similar to that of IP address fragmentation.

20 Setup the Tables:

25 The SDAMT table is a two-dimension table with the Source Address Sequence Number (SASN) as the X-axle index and the Destination Address Sequence Number (DASN) as the Y-axle index; by retrieving this table, we can find the Address Group Number (AGN).

The SDPMT table is also a two-dimension table with the Source Port Sequence Number (SPSN) as the X axle index and the Destination Port Sequence Number (DP SN) as the Y-axle index; Similarly we can got the Port Group Number (PGN).

30 The PMT is a two-dimension table with the Address Group Number (AGN) as the X-axle index and Port Group Number ~GN) as the Y-axle index. From this table, we can ultimately find the policy entry.

35 All these 3 tables have a size of 1024*1024 Words so that it can support up to 1024 IP address fragmentation, 1024 port number fragmentation and 1024 policy entries.

Initially each entry of these three tables is marked as the mode UNUSED, Then a lookup process is carried in the policy list entry by entry to fill up these three tables. A very important principle in this process is that only the UNUSED entry is entered with a replaced entry. If an entry in the table is filled with an exiting entry, then the entry is not replaced.

A policy counter is maintained. Initially it is set to zero. Each time when a new entry is processed in the policy list this counter is increased by one.

A Policy entry can be represented as following:

(<Dest. subnet, Source subnet>, <Dest. port group, Source port group>, protocol type) -->Action

For the protocol type, there are two choices TCP/IP or UDP/IP. These choices are addressed separately unrelated to this invention, but also can be handled in the same way by indexing as disclosed in this invention. For the sake of clarity, these parameters are not further described in the following descriptions:

To fill up the tables, the following steps are processed:

- 1) Get SASNs according to the Source subnet address.
- 2) Get DASNs according to the destination subnet address.
- 3) Get SPSNs according to the source port group.
- 4) Get DPSNs according to the destination port group.
- 5) Using each (SASN, DASN) pair as the index, find the entry position in the SDAMT table, write the policy counter to these position if its status are UNUSED; record all these entry numbers (which you just write or already exist before your writing) to an AGN set.
- 6) Using each (SPSN, DPSN) pair as the index, find the entry position in the SDPMT table, write the policy counter to these position if the status are UNUSED; record all these entry numbers (which you just write or already exist before your writing) to a PGN set.
- 7) For each element AGN belongs to AGN set and each element PGN belongs to PGN set, we combine them to form a policy index set: (PGN, AGN). Then by using each of these pair as the index, find the entry position in the PMT

table, write the policy counter to these positions if the status are UNUSED.

8) Get the next policy entry from the Policy List, go to step 1.

Usage of the Table

5

- 1) Parse the header of the incoming packet, get DA, SA, DP, SP.
- 2) Travel binary tree to get the DA and SA's Address Sequence Number (DASN and SASN).
- 3) Table lookup to get the DP and SP's Port Sequence Number (DPSN and SPSN).
- 4) Lookup table DSAMP to get Address Group Number (AGN) by using DASN and SASN.
- 5) Lookup table DSPMT to get Port Group Number (1)GN by using DPSN and SPSN.
- 6) Lookup table PMT to get the policy number by using AGN and PGN.

10

15

20

25

30

35

A method for processing a policy-lookup for network protection by employing a policy table comprising a plurality of policy-table entries PTE(ip), where ip= 1, 2, 3, ...N and N is a positive integer representing a total number of the PTE(ip), with each PTE(ip) comprising data for defining a plurality of destination address ranges between a first destination address DA1(ip) and a second destination address DA2(ip), a source address ranges between a first source address SA1(ip) and second source address SA(ip), a destination port group ranging between a first destination port DP1(ip) and second destination port DP2(ip) and a source port group ranging between a first source port SP1(ip) and a second source port SP2(ip), the method comprising steps of A) generating an array of destination address segments by arranging ranges represented by {DA1(ip), DA2(ip)}, for ip=1, 2, 3, ...N, according to a destination address sequential order thus generating a plurality of destination address segments S1(Idas) between first destination address A11(Idas) and second destination address A12(Idas) where Idas is a series of destination address sequence number (DASN) and Idas=1, 2, 3, ...IIdas, and IIdas is a positive integer less than or equal to 2N-1. B) Generating an array of source address segments by arranging ranges represented by {SA1(ip), SA2(ip)}, for ip=1, 2, 3, ...N, according to a source address sequential order thus generating a plurality of source address segments S2(Isas) between a first source address A21(Isas) and a

second source address $A22(Isas)$, where $Isas$ is a series of source address sequence number (SASN) and $Isas=1, 2, 3, \dots, IIsas$, and $IIsas$ is a positive integer less than or equal to $2N-1$. C) Forming a source-destination address mapping table (SDAMT) comprising a plurality of SDAMT table entries $SDA(Isas, Idas)$ with $Isas=1, 2, 3, \dots, IIsas$, and $Idas=1, 2, 3, \dots, IIDas$ and $SD(Isas, Idas)=ip1$ wherein $ip1$ is a policy-table entry counter of a first policy table entry wherein the $S2(Isas)$ is included a range defined by $SA1(ip1)$ and $SA2(ip1)$, and the $S1(Idas)$ is included in a range defined by $DA1(ip1)$, $DA2(ip1)$. D) Generating an array of destination port segments by arranging ranges represented by $\{DP1(ip), DP2(ip)\}$, for $ip=1, 2, 3, \dots, N$, according to a destination address sequential order thus generating a plurality of destination address segments $P1(Idps)$ between a first destination port $P11(Idps)$ and a second destination port $P12(Idps)$, where $Idps$ is a series of destination port sequence number (DPSN) and $Idps=1, 2, 3, \dots, IIDps$, and $IIdps$ is a positive integer less than or equal to $2N-1$. E) Generating an array of source port segments by arranging ranges represented by $\{SP1(ip), SP2(ip)\}$, for $ip=1, 2, 3, \dots, N$, according to a source address sequential order thus generating a plurality of source address segments $S2(Isps)$ between a first source port $P21(Isps)$ and a second source port $P22(Isps)$, where $Isps$ is a series of source address sequence number (SPSN) and $Isps=1, 2, 3, \dots, IIsps$, and $IIsps$ is a positive integer less than or equal to $2N-1$. And F) Forming a source-destination port mapping table (SDPMT) comprising a plurality of SDPMT table entries $SDP(Isps, Idps)$ with $Isps=1, 2, 3, \dots, IIsps$, and $Idps=1, 2, 3, \dots, IIDps$ and $SDP(Isps, Idps)=ip2$ wherein $ip2$ is a policy-table entry counter of a first policy table entry wherein the $S2(Isps)$ is included a range defined by $SP1(ip2)$ and $SP2(ip2)$, and the $S2(Idps)$ is included in a range defined by $DP1(ip2)$, $DP2(ip2)$. In a preferred embodiment, the method further includes a step of forming a policy mapping table by generating a policy-mapping table entry $PMT(ip, ip)$ for $ip=1, 2, 3, \dots, N$, wherein $PMT(ip3, ip4)=ip$ for $ip=1, 2, 3, \dots, N$ and $ip3=ip1(R1)$, and $ip4=ip2(R2)$, and $ip1(R1)$ representing all policy-table entry counters in the SDAMT within a two-dimensional range defined by $\{SA1(ip), SA2(ip)\}$ and $\{DA1(ip), DA2(ip)\}$, and $ip2(R2)$ representing all policy-table entry counters in the SDPMT within a two-dimensional range defined by $\{SP1(ip), SP2(ip)\}$ and $\{DP1(ip), DP2(ip)\}$. In a preferred embodiment, the method further includes a step of forming a destination address binary tree by generating an array of tree elements each having a root destination-address and two branch destination addresses and recursively each root destination address is further

assigned as a next level root destination address for generating two next-level branch destination addresses wherein a first root address is $A11(R1)$ where $R1 = N/2$ if N is an even number and $R1$ is $(N+1)/2$ if N is an odd number, and the two branch destination addresses are $A12(R1-1)$ and $A12(R1)$. Forming a source address binary tree by generating an array of tree elements each having a root source-address and two branch destination addresses and recursively each root destination address is further assigned as a next level root destination address for generating two next-level branch destination addresses wherein a first root address is $A21(R1)$ and the two branch destination addresses are $A22(R1-1)$ and $A22(R1)$.

Forming a destination port binary tree by generating an array of tree elements each having a root destination-port and two branch destination ports and recursively each root destination port is further assigned as a next level root destination port for generating two next-level branch destination port wherein a first root address is $P11(R1)$ and the two branch destination ports are $P12(R1-1)$ and $P12(R1)$. And, forming a source port binary tree by generating an array of tree elements each having a root source-port and two branch source ports and recursively each root source port is further assigned as a next level root source port for generating two next-level branch source port wherein a first root address is $P21(R1)$ and the two branch destination ports are $P22(R1-1)$ and $P22(R1)$. In a preferred embodiment, the method further includes a step of receiving an incoming packet containing data for parsing a designated destination and source addresses represented by DDA and DSA respectively, and a designated destination and source ports represented by DDP and DSP respectively. And, searching along the destination address binary tree for determining a destination address root DAR and a destination address branch DAB wherein $DAB < DDA < DAR$ and determining a destination address sequence number $DASN(DDA)$ for the DDA. Searching along the source address binary tree for determining a source address root SAR and a source address branch SAB wherein $SAB < DSA < DAR$ and determining a source address sequence number $SASN(DSA)$ for the DSA. Searching along the destination port binary tree for determining a destination port root DPR and a destination port branch DPB wherein $DPB < DDP < DPR$ and determining a destination port sequence number $DPSN(DDP)$ for the DDP. Searching along the source port binary tree for determining a source port root SPR and a source port branch SPB wherein $SPB < DSP < DPR$ and determining a source port sequence number $SPSN(DSP)$ for the DSP. And, applying the $DASN(DDA)$, $SASN(DSA)$, $DPSN(DDP)$, and

SPSN(DSP) for search the SDAMT, SDPMT, and PMT for finding a policy table entry counter ip for receiving the incoming packet only when a policy-table entry counter ip is found from the PMT.

5 According to the above descriptions, this invention discloses a database for use in processing a table wherein the table including a plurality of table entries each assigned with an ordered table entry counter ip and each entry providing data for defining a plurality of multiple-dimensional spaces. The database includes an array of ordered spatial ranges, e.g., destination and source address and port ranges, 10 each assigned with an ordered spatial range number, e.g., SASN and DASN, generated from fragmenting the multiple-dimensional spaces into the array of order spatial ranges. The database further includes a multiple-dimensional table, e.g., SDAMT or SDPMT. The table is generated from forming a plurality of multiple-dimensional range-spaces by employing the sequential spatial range-numbers as 15 coordinates and assigning an associated table entry counter ip to each block defined by the spatial range-number coordinates for providing an index for correlating each of the sequential spatial range-numbers to the each of the table entry.

Performance Evaluation

20 Assume there are 1024 IP fragmentation and port number fragmentation, there is also a Policy List of 1024 entries.

25 A balanced binary tree is used to hold all the boundary of IP segments, then the height of the tree should be 9. To travel two of these trees we need total of 18 times compare and branch.

30 Since direct table lookup is applied to determine the port segment, only 2 times memory access is needed.

 There are three times table lookup and that requires three times of memory access operations.

35 Totally, 18 compare and branch operations are performed and +5 table lookup operations are carried out.

The computational complexity of policy lookup is reduced from $O(n)$ to $O(\lg n)$, where the n is the length of the Policy List.

Although the present invention has been described in terms of the presently preferred embodiment, it is to be understood that such disclosure is not to be interpreted as limiting. Various alterations and modifications will no doubt become apparent to those skilled in the art after reading the above disclosure. Accordingly, it is intended that the appended claims be interpreted as covering all alterations and modifications as fall within the true spirit and scope of the invention.

5
10

11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199